

A Secure Proxy Signature Scheme with Fault Tolerance Based On Discrete Logarithm Problem

H. Elkamchouchi¹, Heba G. Mohamed², Fatma Ahmed³ and Dalia H. ElKamchouchi⁴

¹Dept. of Electrical engineering, Faculty of Engineering, Alexandria University, Egypt

helkamchouchi@ieee.org

²Dept. of Electrical engineering, Arab Academy for Science and Technology (AAST),

Egyptheba.g.mohamed@gmail.com

³Dept. of Electrical engineering, Faculty of Engineering, Alexandria University, Egypt

moonyally@yahoo.com

⁴Dept. of Electrical engineering, Faculty of Engineering, Alexandria University, Egypt

Daliakamsh@yahoo.com

ABSTRACT: The techniques of proxy signature and fault tolerance are two important issues in modern communication. Proxy signature scheme permits an original signer to delegate his/her signing capability to a proxy signer, and then the proxy signer generates a signing message on behalf of the original signer. To communicate securely over an unreliable public network, the two parties must be able to authenticate one another and agree on a secret encryption key. Authenticated key agreement protocols have an important role in building a secure communications network between the two parties. In this paper, we propose a secure proxy signature scheme with fault tolerance over an efficient and secure authenticated key agreement protocol based on the discrete logarithm problem. The scheme does not require any extra mechanism, such as checkpoints, to achieve the property of fault tolerance.

KEYWORDS: Discrete logarithm, Fault tolerance, Key agreement, Proxy signature.

I. INTRODUCTION

Due to the rapid growth in modern communication systems, fault tolerance and data security are two important issues in a secure transaction. During the transmission of data between the sender and receiver, errors may occur frequently. Therefore, the sender must re-transmit the data to the receiver in order to correct these errors, which makes the system very feeble. Digital signature schemes with fault tolerance make it possible for error detections and corrections during the processes of data computations and transmissions. Previously, Zhang [1] and Lee and Tsai [2] have respectively proposed two efficient fault-tolerant schemes based on the RSA cryptosystem. Both of them can efficiently check the sender's identity and keep the confidentiality of the transmitted document. Furthermore, they can detect the errors and correct them. However, these schemes have a common weakness in security. Huifang Xue [3] has improved the mechanism of Lee and Tsai by providing extra security against Chosen Ciphertext Attacks (CCA) using a permutation matrix. If a malicious looks into the message he will find it difficult to understand or calculate checksum/ hash value due to the randomization of permutation matrix.

Proxy signature scheme was first presented by Mambo et al. [4] in 1996. Their proxy signature scheme allows an original signer to delegate his/her signing right to a proxy signer to sign the message on behalf of an original signer. Later, the verifier, which knows the public keys of the original signer and a proxy signer can check a validity of a proxy signature issued by a proxy signer. The classification of the proxy signature is dependent on the basis of delegation, namely full delegation, partial delegation, and delegation by warrant, and presents a well-organized strategy. In the full delegation, the proxy signer signs document using the same secret key by the original signer. The drawback of proxy signature with the full delegation is the difficulty to distinct/differentiate between the original signer and the proxy signer.

In the partial delegation, the proxy key is derived from the secret key of the original signer and hands it over to the proxy signer as a delegation capability. Due to the partial delegation, the proxy signer's signing capability cannot be restricted, so he/she can misuse the delegation capability. The weaknesses of full delegation and partial delegation are eliminated by the partial delegation with warrant. A warrant explicitly states the signer's identity, delegation period, and the qualification of messages on which the proxy signer can sign period and the types of a message on which a proxy signer can sign.

There are two types of partial delegation; with warrant protected and unprotected proxy signature

schemes. In the unprotected proxy signature scheme, a proxy signature is generated by both proxy signer and original signer. In this case, the verifier cannot distinguish the identity of a signer. In the protected proxy signature scheme, a proxy signature is generated by the proxy signature key of an original signer and also with a private key of a proxy signer. In 1997, Kim et al. [5] proposed a scheme using the concept of partial delegation with a warrant to restrict the proxy signer signing capability. In 1999, Okamoto et al. [6], for the first time, proposed a proxy unprotected signature scheme based on RSA scheme. A proxy-protected signature scheme based on the RSA assumption was proposed by Lee, et al. in 2001 [7], [8]. In 2009, Shao [9] proposed the proxy-protected signature scheme based on RSA. In 2011, Popescu [10] introduced a secure proxy signature scheme with delegation by warrant, and the scheme is based on the difficulty of solving the discrete logarithm problem (DLP).

The two parties must authenticate one another and agree on a secret encryption key to communicate together securely over an unreliable public network. To achieve this, key establishment protocols are applied at the beginning of a communication session in order to verify the identities of both parties and build a common session key. Authenticated key agreement protocols have an important role in establishing secure communications between the two parties over the open network. The most famous protocol for key agreement was proposed by Diffie and Hellman which is based on the concept of public-key cryptography (DL) [11]. There are two types of the Diffie-Hellman protocol, namely static and ephemeral. In the first one, the parties exchange static public keys, and in the second, they exchange ephemeral public keys [12]. The important feature of the designed protocol is that the established session key is formed as a combination of static and ephemeral private keys of two parties.

This paper demonstrates the effect of an efficient and secure authenticated key agreement protocol on a proxy protected signature scheme with fault tolerance based on DLP. The designed protocol for the authenticated key agreement is secure, efficient, and provides authentication between two entities before exchanging the session keys. The remaining parts of this paper are organized as follows: In Section II, we elaborate security properties of the proxy signature scheme. Next, we discuss the designed protocol in Section III. In Section IV, we discuss Iuon and Chin Chang's Scheme. In section V, we proposed our scheme. We analyze the security properties and common attacks of our proposed scheme in Section VI. Finally, in Section VII, we give our conclusion.

II. SECURITY REQUIREMENTS OF PROXY SIGNATURE

The security requirements for any proxy signature are first studied in [4] and later were improved in [7], [8]. According to them, a secure proxy signature scheme is expected to satisfy the following five requirements:

1. **Verifiability:** A verifier can be confident of the original signer's agreement on the signed message from a proxy signature.
2. **Strong unforgeability:** Only the designated proxy signer can generate a valid proxy signature.
3. **Strong identifiability:** The identity of the proxy signer can be determined by any verifier from a proxy signature.
4. **Strong undeniability:** The proxy signer cannot repudiate the signature creation against anyone else, once he/she creates a valid proxy signature on behalf of an original signer.
5. **Prevention of misuse:** The responsibility of the proxy signer should be determined explicitly if he/she misuses the proxy key for the purposes other than generating a valid proxy signature

III. NEW KEY AGREEMENT PROTOCOL

The used protocol for authenticated key agreement [10] provides authentication between the two parties A and B before exchanging the session keys. The protocol consists of three phases; The Registration Phase, The Transfer and Substantiation Phase, and The Key Generation Phase. Fig. 1 shows the overall operation of the new protocol. The system picks short-term private key r_A, r_B , they are random integers $2 \leq r_A, r_B < p1$ and $\text{GCD}(r, p1) = 1$. $p1 = (p - 1)$ where p is a large safe prime $p = n'p' + 1$ (n' is a small prime number, usually taken by 2 and p' is a large prime number usually at least 1024 bits). t_A, t_B are short-term public keys where $t_A = g^{r_A} \bmod p$ and $t_B = g^{r_B} \bmod p$. g is a generator of \mathbb{Z}_p^* . Furthermore, the system picks long-term private keys x_A, x_B they are random integer where $2 \leq x_A, x_B < p1$ and $\text{GCD}(x, p1) = 1$ then, computes long-term public key y_A, y_B where $y_A = g^{x_A} \bmod p$ and $y_B = g^{x_B} \bmod p$, K_{AB} is the shared secret key calculated by the new secure protocol between the two parties A and B.

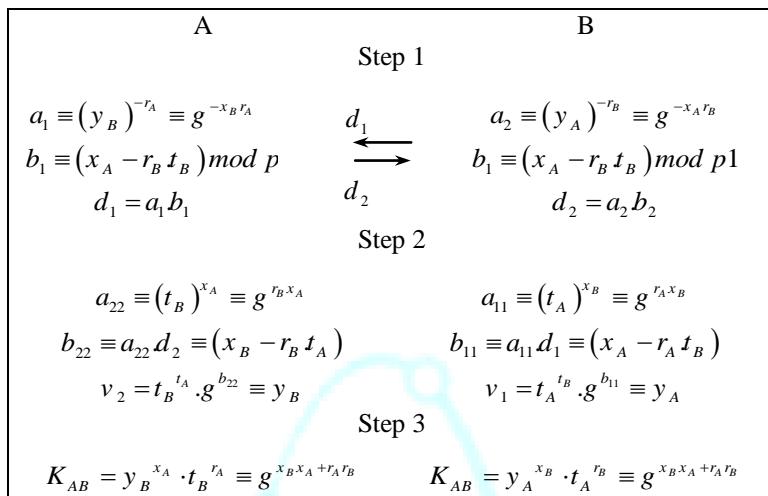


Fig.1. Overall operation of the proposed protocol

In the first step, the number of scalar multiplications required is one, the number of exponentiation required is one, and the total number of sending message is one. In the second step, each user will be verified from the other one because in the first step each user uses the short-term private key which belongs to him/her in calculation.

IV. IUONAND CHIN CHANG'S SCHEME

Iuon and Chin Chang's scheme [15] is developed from the concept of meta-ElGamal signature scheme [14] and the concept of Zhang's fault-tolerant signature scheme. In ElGamal digital signature scheme, a system first chooses a large prime p and a generator g , such that $g \in \mathbb{Z}_p^*$ with order $p - 1$. Both p and g can be shared among a system of users. To generate a key pair, the signer A first chooses a random number x_A , $x_A \in \mathbb{Z}_{p-1}$ and calculates $y_A = g^{x_A} \bmod p$. A keeps x_A secret and publishes y_A . Suppose that the signer Alice will send a message with her signature to the receiver Bob. Alice possesses a secret key x_A and a public key y_A . The proposed scheme can be divided into two procedures:

1. The signature generation procedure.
2. The fault tolerance and signature verification procedure.

4.1 The Signature Generation Procedure

1. Alice first divides the transmitted message M into numerical 3×3 message matrices X_l 's, such that

$$X_l = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \quad (1)$$

Where m_{ij} , $1 \leq i \leq 3, 1 \leq j \leq 3$, is a message block and $m_{ij} \in \mathbb{Z}_{p-1}$

2. For each message matrix X_l , Alice calculates its signature and constructs an expand matrix D_l , such that

$$D_l = \begin{bmatrix} m_{11} & m_{12} & m_{13} & r_1 & s_1 & t_1 \\ m_{21} & m_{22} & m_{23} & r_2 & s_2 & t_2 \\ m_{31} & m_{32} & m_{33} & r_3 & s_3 & t_3 \\ r^1 & r^2 & r^3 & & & \\ s^1 & s^2 & s^3 & & & \\ t^1 & t^2 & t^3 & & & \end{bmatrix} \quad (2)$$

The r_i, s_i, t_i, r^i, s^i and t^i can be calculated by using the following equations

$$r_i = g^{k_i} \bmod p, \quad (3)$$

$$t_i = \sum_{j=1}^3 m_{ij} \bmod p - 1, \quad (4)$$

$$s_i = (H(m_{i1}) \cdot t_i - H(m_{i2}) \cdot r_i \cdot x_A) (H(m_{i3}) \cdot k_i)^{-1} \bmod p - 1, \quad (5)$$

$$r^j = g^{k_j} \bmod p, \quad (6)$$

$$t^j = \sum_{i=1}^3 m_{ij} \bmod p - 1, \quad (7)$$

$$s^j = (H(m_{1j}) \cdot t^j - H(m_{2j}) \cdot r^j \cdot x_A) (H(m_{3j}) \cdot k^j)^{-1} \bmod p - 1, \quad (8)$$

where $H()$ is a public one-way hash function.

4.2 The Fault Tolerance and Signature Verification Procedure

1. Bob first detects errors by checking the equations

$$t_i = \sum_{j=1}^3 m_{ij} \bmod p \quad \text{and} \quad t^j = \sum_{i=1}^3 m_{ij} \bmod p \quad (9)$$

If there is an error in m_{uv} , $1 \leq u, v \leq 3$, we must have that $t_u \neq \sum_{j=1}^3 m_{uj} \bmod p - 1$ and $t^v \neq \sum_{i=1}^3 m_{iv} \bmod p - 1$

Therefore, the error could be easily detected.

2. After the error is detected in m_{uv} , it may be corrected by using either one of the following two equations

$$m_{uv} = t_u - \sum_{j \neq v} m_{uj} \bmod p \quad (10)$$

$$m_{uv} = t^v - \sum_{i \neq u} m_{iv} \bmod p$$

3. After correcting the errors, Bob has to verify the validity of the recovery and its corresponding signatures by checking whether

$$g^{H(m_{i1})t_i} = y_A^{H(m_{i2}) \cdot r} \cdot r_i^{H(m_{i3}) \cdot s_i} \bmod p$$

$$g^{H(m_{1j})t^j} = y_A^{H(m_{2j}) \cdot r^j} \cdot r^j^{H(m_{3j}) \cdot s^j} \bmod p \quad (11)$$

or not. If the above verifications are positive, Bob will believe that the contents of the recovered messages are valid. Otherwise, Bob can choose not to accept the received messages.

V. THE PROPOSED PROXY SIGNATURE SCHEME

The proposed proxy scheme is focused on the proxy protected proxy signatures with the new authenticated key agreement protocol with fault tolerance based on the DLP. The system is divided into four phases: System setup, Proxy key generation, Proxy key verification, Proxy signature generation and Fault tolerance and Proxy signature verification.

5.1 System Setup

It is supposed that the original signer A invites the proxy signer P to perform signing on behalf of him/her, and the verifier or the receiver B verifies the validity of the generated signature. Also, suppose that p is a large prime number, and g is a generator for Z_p^* . ID_A and ID_P are the identity of the original signer and the proxy signer, respectively. $x_A, x_P \in Z_p^*$ are the private key of the original signer and the proxy signer, respectively, then compute public key y_A and y_P where, $y_A = g^{x_A} \bmod p$ and $y_P = g^{x_P} \bmod p$ are the public keys of the original signer and proxy signer, respectively. K_{AP} is the shared secret key between A and P

5.2 Proxy Key Generation

1. The original signer entity A first divides the transmitted message M into numerical 3×3 message matrices and do the following:

- Selects an arbitrary integer value $k_i, k^j \in Z_{p-1}$
- Find $r_i = g^{k_i} \bmod p$ and $r^j = g^{k^j} \bmod p$
- Calculate warrant m_w where, m_w must be created from ID_A, ID_P and other data on the delegation.
- Compute $h(m_w \square r_i)$ and $h(m_w \square r^j)$
- Find $\sigma_i = k_i + x_A * (h(m_w \square r_i) \oplus K_{AP}) \bmod p - 1$, $\sigma^j = k^j + x_A * (h(m_w \square r^j) \oplus K_{AP}) \bmod p - 1$ for all $1 \leq i \leq 3, 1 \leq j \leq 3$.
- Send $(m_w, r_i, r^j, K_{AP}, \sigma_i, \sigma^j)$ to the proxy signer in the secure channel.

2. The proxy signer checks the validity of $(m_w, r_i, r^j, K_{AP}, \sigma_i, \sigma^j)$ by verifying whether or not the following equation holds $g^{\sigma_i} \equiv r_i y_A^{h(m_w \square r_i) \oplus K_{AP}}$ and $g^{\sigma^j} \equiv r^j y_A^{h(m_w \square r^j) \oplus K_{AP}}$. If the verification is successful, the proxy signer then computes an alternative proxy private/public key pair σ_{pr} and y_{pr} , respectively, such that

$$\sigma_{i_{pr}} = \sigma_i + x_P * (h(m_w \square r_i) \oplus K_{AP}) \bmod p - 1$$

$$y_{i_{pr}} = g^{\sigma_{i_{pr}}} \bmod p \quad (12)$$

$$\begin{aligned} \sigma_{pr}^j &= \sigma^j + x_P * (h(m_w \square r^j) \oplus K_{AP}) \bmod p - 1 \\ y_{pr}^j &= g^{\sigma_{pr}^j} \bmod p \end{aligned} \quad (13)$$

5.3 Signature Generation

Now, the proxy signer P will sign a message M on behalf of the original signer, he uses σ_{pr} to perform a signing operation. The proxy signature on the message M is as follows

$$\begin{aligned} s_i &= (H(m_{i1}) \cdot t_i - H(m_{i2}) \cdot \sigma_i \cdot y_{i_{pr}}) (H(m_{i3}) \cdot \sigma_{i_{pr}})^{-1} \bmod p - 1, \\ s^j &= (H(m_{1j}) \cdot t^j - H(m_{2j}) \cdot \sigma^j \cdot y_{pr}^j) (H(m_{3j}) \cdot \sigma_{pr}^j)^{-1} \bmod p - 1, \end{aligned} \quad (14)$$

Where $t_i = \sum_{j=1}^3 m_{ij} \bmod p - 1$ and $t^j = \sum_{i=1}^3 m_{ij} \bmod p - 1$, For all $1 \leq i \leq 3, 1 \leq j \leq 3$

5.4 Fault tolerance and Signature Verification

1. The receiver B first detects errors by checking the equations

$$t_i = \sum_{j=1}^3 m_{ij} \bmod p \quad \text{and} \quad t^j = \sum_{i=1}^3 m_{ij} \bmod p \quad (15)$$

If there is an error in m_{uv} , $1 \leq u, v \leq 3$, we must have that $t_u \neq \sum_{j=1}^3 m_{uj} \bmod p - 1$ and $t^v \neq \sum_{i=1}^3 m_{iv} \bmod p - 1$

.Therefore, the error could be easily detected.

2. After the error is detected in m_{uv} , it may be corrected by using either one of the following two equations

$$\begin{aligned} m_{uv} &= t_u - \sum_{j \neq v} m_{uj} \bmod p \\ m_{uv} &= t^v - \sum_{i \neq u} m_{iv} \bmod p \end{aligned} \quad (16)$$

3. The receiver B receive the signed message and he has to check whether or not the following equations hold:

$$\begin{aligned} g^{H(m_{i1}) \cdot t_i} &= y_{i_{pr}}^{H(m_{i3}) \cdot s_i} \cdot (r_i \cdot y_A^{h(m_w \cdot r_i)})^{H(m_{i2}) \cdot y_{i_{pr}}} \bmod p \\ g^{H(m_{1j}) \cdot t^j} &= y_{pr}^{H(m_{3j}) \cdot s^j} \cdot (r^j \cdot y_A^{h(m_w \cdot r^j)})^{H(m_{2j}) \cdot y_{pr}^j} \bmod p \\ y_{pr}^j &= r (y_A y_P)^{h(m_w \square r) \oplus K_{AP}} \bmod p \end{aligned} \quad (17)$$

VI. SECURITY ANALYSIS

In the following, we show that the proposed schemes satisfy the security features, namely, verifiability, strong unforgeability, strong, undeniability, strong identifiability, and prevention of misuse.

6.1 Verifiability

According to the step 1 of the fault tolerance and the signature verification procedure, if an error occurs in m_{ij} , therefore $t_i \neq m_{i1} + m_{i2} + m_{i3} \bmod p$ and $t^j \neq m_{1j} + m_{2j} + m_{3j} \bmod p$. The fault message m_{ij} can be recovered by computing, if the rest of the messages m_{ik} 's where $k = 1 \text{ to } 3$ and $k \neq j$, in the i^{th} row are correct. On the otherhand, if the rest of the messages m_{kj} 's, where $k = 1 \text{ to } 3$ and $k \neq i$, in the j^{th} column are correct, the fault message m_{ij} also can be recovered by computing $m_{ij} = t^j - (\sum_{k=1 \text{ to } 3, k \neq i} m_{kj}) \bmod p$. Therefore, an error is correctable only when no other errors simultaneously occur in the same row i and the same column j . In the proposed scheme, we can correct four errors in a message matrix X at most. Figure 2 illustrates the correctable conditions when four errors simultaneously occur in a message matrix. Therefore, all the four errors can be corrected by using the check-sums in either the row or the column direction.

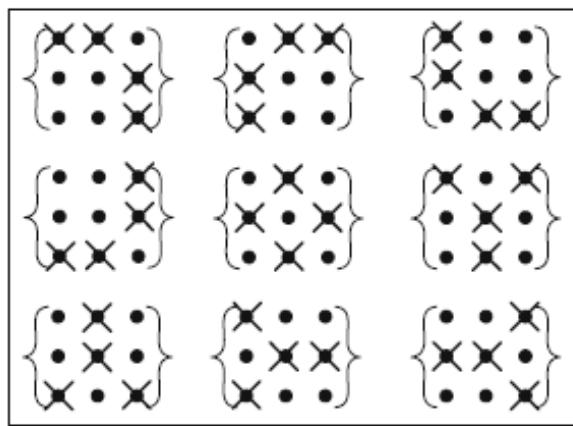


Fig. 2. The correctable conditions when there are four errors simultaneously occurring in a message matrix

According to the step 2, the receiver B can check the verification equation:

$$\begin{aligned}
y_{pr} &= g^{\sigma_{pr}} \bmod p \\
&= g^{\sigma + x_p * (h(m_w \square r) \oplus K_{AP})} \bmod p \\
&= g^{\sigma} g^{x_p * (h(m_w \square r) \oplus K_{AP})} \bmod p \\
&= g^{k + x_A * (h(m_w \square r) \oplus K_{AP})} g^{x_p * (h(m_w \square r) \oplus K_{AP})} \bmod p \\
&= g^k g^{x_A * (h(m_w \square r) \oplus K_{AP})} g^{x_p * (h(m_w \square r) \oplus K_{AP})} \bmod p \\
&= g^k (g^{x_A} g^{x_p})^{h(m_w \square r) \oplus K_{AP}} \bmod p \\
&= r(y_A y_p)^{h(m_w \square r) \oplus K_{AP}} \bmod p
\end{aligned}$$

6.2 Strong Unforgeability

In this scheme, the proxy signature is created with the proxy signer's secret key x_p and delegated proxy key σ . The proxy key is bound with the original signer's secret key x_A and the session key K_{AP} . No one (including the original signer) can construct the proxy signature. If the original signer tries to construct the proxy private key from a proxy public key, he/she will need to solve the DLP. However, the DLP is difficult. Moreover, the verification of $h(m_w \square r) \oplus K_{AP}$ with the signed message prevents the dishonest party from the creation of forged proxy signatures. Therefore, any party, including the original signer cannot forge a valid proxy signature, and thus the proposed scheme satisfies the unforgeability property.

6.3 Strong Identifiability

Any verifier can determine the identity of the proxy signer from the proxy signatures created by the proxy signer. Therefore, in the proposed scheme, any verifier can identify the identity of the proxy signer from the proxy signature generated by him on the message M .

6.4 Strong Undeniability:

In the proposed scheme, the involvements of both original signer and proxy signer are determined by the secret keys x_p and x_A from the proxy signature. Thus, the proxy signer and the original signer cannot deny their involvement in a valid proxy signature. Consequently, the scheme satisfies the undeniability property.

6.5 Prevention of Misuse

In the proposed scheme, the proxy signer cannot forge the delegated rights. The responsibility of the proxy signer is determined from the warrant m_w in the case of the proxy signer's misuse. Therefore, the original signer's misuse is also prevented because he cannot compute a valid proxy signature against the proxy signer. Next, we show that our scheme is heuristically secured by considering the following five most common attacks.

Known-Key Security (K-KS): In the proposed scheme, if an established session key between original signer and proxy signer is disclosed, the adversary is unable to learn the other established session keys. In each run of the proposed scheme between the two parties, a unique session key which depends on r_A and r_p should be produced. Therefore, the adversary cannot compute K_{AP} and cannot calculate $\sigma = k + x_A * (h(m_w \square r) \oplus K_{AP}) \bmod p - 1$.

(Perfect) Forward Secrecy: If both secret keys of two parties are compromised, the adversary is unable to derive the old session keys established by two parties. The protocol also possesses forward secrecy. Suppose that adversary compromises the private keys x_A and he/she cannot calculate $\sigma = k + x_A * (h(m_w \square r) \oplus K_{AP}) \bmod p - 1$. However, the secrecy of previous session keys established by the honest parties is not affected, because an adversary who captured the private key x_A should extract the ephemeral keys r_A or r_p from the exchanged values to know the previous or next session keys between them. Thus, he/she still fails to produce σ send to proxy signer. However, this is DLP.

Key-Compromise Impersonation (K-CI): When the private key of original signer is compromised, it may be desirable that this event does not enable an adversary to impersonate other entities to the original signer. Suppose that x_A is disclosed. Now an opponent who knows this value can clearly impersonate the original signer. In the proposed scheme, the opponent cannot impersonate the proxy signer to the original signer and compute $\sigma_{pr} = \sigma + x_p * (h(m_w \square r) \oplus K_{AP}) \bmod p - 1$ without knowing the proxy signer's private key x_p . From the success of the impersonation, the opponent must know the original signer's ephemeral key r_A . So, in this case, the opponent should extract the value r_A from $t_A \equiv g^{r_A} \bmod n$; however, he/she cannot calculate the sharing key, this is DLP.

Unknown Key-Share (UK-S): The original signer A cannot be coerced into sharing a key with the proxy signer P without the knowledge of the original signer, i.e., A believes that the key is shared with some entity $C \neq P$, and P believes that the key is shared with A . The used protocol prevents unknown key-share. Corresponding to the proxy signer's public static and ephemeral keys y_p, t_p , an adversary cannot register proxy signer's public keys y_p, t_p as its own, and according to the assumption of this protocol that d_2 has verified that P possesses the private static and ephemeral keys x_p, r_p , respectively. So an adversary cannot deceive the original assuming that $\sigma_{pr} = \sigma + x_p * h(m_w \square r) \oplus K_{AP} \bmod p - 1$ was originated from him. Therefore, the original signer cannot be coerced into sharing K_{AP} with the proxy signer without his/her knowledge.

VII. CONCLUSION

In this paper, we considered the problem of combining proxy protected signature scheme and fault tolerance with a new key agreement protocol based on DLP. Our scheme does not consider the proxy revocation mechanism. The proposed scheme satisfies the capability of correcting four at most errors for each 3×3 message matrix. On the other hand, the scheme satisfies the necessary security requirements of proxy signature and has a

secure channel to deliver the proxy key, through the designed new protocol that meets the security attributes under the assumption of DLP.

REFERENCES

- [1] C.N. Zhang, "Integrated Approach for Fault Tolerance and Digital Signature in RSA," IEEE Proceedings-Computers & Digital Techniques, vol. 146, no. 3, pp. 151-159, 1999
- [2] N. Lee and W. Tsai, "Efficient Fault-tolerant Scheme basde on the RSA system," IEEE Proceedings – Computer and Digital Techniques, vol. 150, no. 1, pp. 17-20, 2003.
- [3] Xue, H. (2010) Improving the Fault-Tolerant Scheme Based on the RSA System. International Symposium on Computational Intelligence and Design, Hangzhou, 29-31 October 2010, 31-33.
- [4] M. Mambo, K. Usuda, E. Okamoto, Proxy signature: delegation of the power to sign the message, IEICE Trans. Fundamentals E79-A (9) (1996) PP. 1338 - 1353.
- [5] S. Kim, S. Park and D. Won, "Proxy signatures", In: ICICS97, LNCS 1334, Springer-Verlag, (1997), pp. 223-232.
- [6] T. Okamoto, M. Tada and E. Okamoto, "Extended proxy signatures for smart card", In: Proceedings of Information Security Workshop 99, LNCS 1729, Springer-Verlag, (1999), pp. 247-258.
- [7] B. Lee, H. Kim and K. Kim, "Secure mobile agent using strong non-designated proxy signature", In: Information security and private (ACISP01), LNCS 2119, Springer-Verlag, (2001), pp. 474-486.
- [8] B. Lee, H. Kim and K. Kim, "Strong proxy signature and its applications", In: Proceeding of the 2001 symposium on cryptography and information security (SCIS01), vol. 2, no. 2, (2001), pp. 603-608
- [9] Hwang S., Chen C.: A new proxy multi- signature scheme, International workshop on cryptology and network security, Tamkang University Taipei, Taiwan, pp 26-28 (2001).
- [10] Constantin Popescu: A Secure Proxy Signature Scheme with Delegation by Warrant, SIC: Volume 20, issue 4, pp. 373- 380 (2011).
- [11] W. Diffie and M. Hellman, "New directions in cryptography," IEEE Transactions on Information Theory, vol. IT-22, no. 6, PP. 644-654, November, 1976.
- [12] K. Chalkias, F. Mpaldimtsi, D. H. Varsakelis, and G. Stephanides, "On the Key-compromise impersonation vulnerability of one-pass key establishment protocols," in Proc. International Conference on Security and Cryptography (SECRYPT 2007), Barcelona, Spain, July 28-31, 2007.
- [13] Fatma Ahmedand Dalia Elkamchouchi," A New Efficient Protocol for Authenticated Key Agreement," IACSIT International Journal of Engineering and Technology, Vol. 2, No. 4, April 2013,pp.510-512.
- [14] P. Horster, M. Michels, and H. Petersen, "Meta-message recovery and meta-blind signature schemes based on the discrete logarithm problem and their applications,"*Lecture Notes in Computer Science-Advance in Cryptology: AsiaCrypt '94*, vol. 917, pp. 224-237, 1995.
- [15] I.C. Lin and C.C. Chang, "An Efficient Fault-Tolerant Digital Signature Scheme Based on the Discrete Logarithm Problem," Autonomic and Trusted Computing, vol. 4158, pp. 601-610, 2006.

IJSTRE